

Title	Asymmetric neighborhood functions accelerate ordering process of self-organizing maps
Author(s)	Ota, Kaiichiro; Aoki, Takaaki; Kurata, Koji; Aoyagi, Toshio
Citation	Physical Review E (2011), 83(2)
Issue Date	2011-02
URL	http://hdl.handle.net/2433/139553
Right	© 2011 American Physical Society
Type	Journal Article
Textversion	publisher

Asymmetric neighborhood functions accelerate ordering process of self-organizing mapsKaiichiro Ota,^{1,*} Takaaki Aoki,¹ Koji Kurata,² and Toshio Aoyagi^{1,3}¹*Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan*²*Faculty of Engineering, University of the Ryukyus, Okinawa 903-0213, Japan*³*JST, CREST, Tokyo 102-0075, Japan*

(Received 27 January 2010; revised manuscript received 9 December 2010; published 4 February 2011)

A self-organizing map (SOM) algorithm can generate a topographic map from a high-dimensional stimulus space to a low-dimensional array of units. Because a topographic map preserves neighborhood relationships between the stimuli, the SOM can be applied to certain types of information processing such as data visualization. During the learning process, however, topological defects frequently emerge in the map. The presence of defects tends to drastically slow down the formation of a globally ordered topographic map. To remove such topological defects, it has been reported that an asymmetric neighborhood function is effective, but only in the simple case of mapping one-dimensional stimuli to a chain of units. In this paper, we demonstrate that even when high-dimensional stimuli are used, the asymmetric neighborhood function is effective for both artificial and real-world data. Our results suggest that applying the asymmetric neighborhood function to the SOM algorithm improves the reliability of the algorithm. In addition, it enables processing of complicated, high-dimensional data by using this algorithm.

DOI: [10.1103/PhysRevE.83.021903](https://doi.org/10.1103/PhysRevE.83.021903)

PACS number(s): 87.19.L-, 05.65.+b, 84.35.+i

I. INTRODUCTION

Topographic mapping is observed in mammalian primary sensory cortices as a topology-preserving correspondence between the feature of external stimuli and the spatial position of the activated neurons [1,2]. A self-organizing map (SOM) learning algorithm was introduced as a simplified neural network model to explain the formation of such cortical topographic maps [3–7]. While the SOM originated as a biophysical model of the topographic mapping seen in the primary sensory cortex, it has found wide applications in data mining and as a visualization method for complex data sets. In this study, we focus on such practical aspects of information processing using the SOM.

A useful feature of the topographic map generated by the SOM learning algorithm is that the dimensionality of the input (stimulus) is reduced when mapped to a spatially low-dimensional array of units. Moreover, it is known that maps generated by the algorithm approximate the probability density of the stimuli in the stimulus space [8,9]. These facts imply that the algorithm can capture essential information in complicated, high-dimensional data sets by mapping them to low-dimensional representation spaces. Thus, the algorithm can be applied to certain types of information processing, such as data mining, visualization, and nonlinear principal component analysis [10].

For the SOM algorithm to be applicable to practical data processing, the algorithm must be able to reliably and quickly form a globally topographic map. However, this process is often disturbed by the emergence of a topological defect in the map [11,12], which then separates the entire map into multiple, locally ordered regions. A map having a topological defect is no longer globally topographic and, thus, does not accurately represent the essential structure of the data set. Once a topological defect appears in a map during the

learning process, it tends to remain for a long time and slows down the ordering process of the SOM. Consequently, the presence of this topological defect tends to prevent the system from reaching a globally ordered state and hinders practical application of the algorithm.

In this study, we investigate a solution against topological defects that appear during learning. Some of the authors showed in a previous study that the use of asymmetric neighborhood functions in the algorithm is effective for removing topological defects and accelerating correct map formation [13]. The neighborhood function determines the spatial dependence of the effective learning rate on the unit array and is conventionally a symmetric function such as a Gaussian function. However, this symmetry tends to stabilize the defect, allowing it to persist for a long time. The previous study mainly dealt with the simplest, one-dimensional case (mapping from a one-dimensional stimulus space to a one-dimensional chain of units), where a topological defect appeared as a kinklike state formed by the reference vectors of the units. In the case of a conventional, symmetric neighborhood function, the defect moves similarly to a random-walk particle. In contrast, the asymmetry of the neighborhood function adds driftlike motion to the defect. As a result, the defect can be moved quickly out of the map. However, under more general conditions, i.e., higher-dimensional stimulus, it is not evident whether the asymmetric neighborhood function is effective in removing topological defects, because the geometric properties and stability of a defect may change depending on the dimensionality of both the stimulus and unit space.

In this paper, we demonstrate that the asymmetric neighborhood function can reliably eliminate topological defects, and that it accelerates the ordering process of the SOM irrespective of the dimensionalities of the stimulus and the unit space. In Sec. II, we describe the SOM algorithm and the asymmetric neighborhood function. In Sec. III, we show the effectiveness of the asymmetric neighborhood function through theoretical arguments and numerical simulations, using various combinations of dimensions for the stimulus

*kaiichiro@acs.i.kyoto-u.ac.jp

space and the array of units. We test the performance of the algorithm on three real-world data sets before concluding the paper in Sec. IV.

II. ALGORITHM

The SOM can generate a globally topographic map that maps a set of D_s -dimensional stimuli to a D_u -dimensional array composed of N units [Fig. 1(a)]. Each unit i has its own preferred stimulus feature represented by the reference vector $\mathbf{m}_i \in \mathbb{R}^{D_s}$ ($i = 1, 2, \dots, N$). An input stimulus or data $\mathbf{x} \in \mathbb{R}^{D_s}$ is mapped to the best-matching unit c according to the winner-take-all rule, given as

$$\|\mathbf{x} - \mathbf{m}_c\| = \min_i \{\|\mathbf{x} - \mathbf{m}_i\|\}, \quad (1)$$

where $\|\cdot\|$ denotes the Euclidean norm. The SOM learning algorithm iteratively updates the reference vectors as follows: At each time step t , the stimulus $\mathbf{x}(t)$ is given. Every reference vector \mathbf{m}_i is then updated as

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + \alpha h(r_{ic})[\mathbf{x}(t) - \mathbf{m}_i(t)] \quad (i = 1, \dots, N), \quad (2)$$

where $\alpha > 0$ is the constant learning-rate parameter. Here, the neighborhood function $h(r)$ represents the spatial dependence

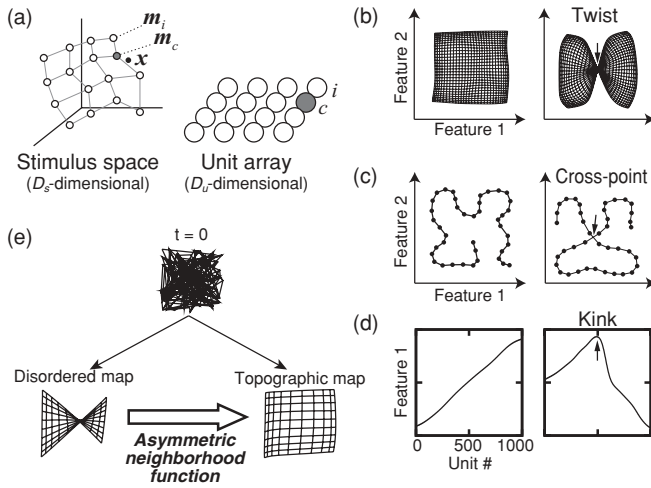


FIG. 1. (a) Schematic of the SOM algorithm. Each unit i has its own preferred stimulus feature, which is indicated by the reference vector \mathbf{m}_i in the stimulus representation space. A stimulus \mathbf{x} is mapped to the best-matching unit c represented by the gray circles. (b), (c), and (d) Examples of globally ordered maps (left) and disordered maps (right) for $(D_s, D_u) = (2, 2)$, $(2, 1)$, and $(1, 1)$, respectively. In figures showing two-dimensional stimuli (b and c), the reference vectors are plotted in the two-dimensional stimulus space, and every pair corresponding to the neighboring units is joined by a line; for one-dimensional stimuli (d), the reference vectors of all units are plotted, with the horizontal axis indicating the index of units. (e) Typical scenarios of the SOM learning algorithm. Topological defects may emerge in the map and slow down the ordering process depending on the stimulus distribution and the initial conditions of the weight vectors. Note that a random initial condition ($t = 0$) is illustrated as an almost entirely disordered state of the map. Use of the asymmetric neighborhood function can prevent emergence of topological defects, resulting in the formation of a globally topographic map.

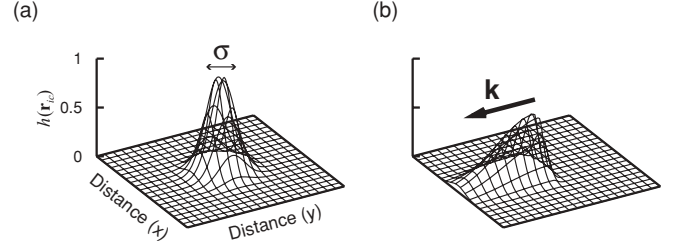


FIG. 2. (a) Symmetric neighborhood function for two-dimensional unit arrays ($D_u = 2$). (b) Asymmetric neighborhood function [Eq. (3) with $\beta = 3$]. Vector \mathbf{k} indicates the direction of asymmetry.

of update intensity, where $r_{ic} \geq 0$ denotes the distance between the updated unit i and the best-matching unit c . We assume the Gaussian function $h(r) = \exp(-r^2/2\sigma^2)$ [Fig. 2(a)], where the parameter σ specifies the learning width on the unit array.

The iteration of the learning step we have described can lead to the formation of a globally topographic map, because each learning step moves the reference vectors toward a configuration that fits the distribution of the given data well. The left panel of Fig. 1(b) depicts a globally topographic map for two-dimensional stimuli and a two-dimensional unit array ($D_s = D_u = 2$). In the figure, the reference vectors of all units are plotted with lines between every pair of nearest-neighbor units. As seen in the figure, for reference vectors close to each other in the stimulus space, their corresponding unit pair is at a close distance on the unit array; this indicates that similar stimuli are mapped to the same unit or neighboring units. Such an ordered configuration of the reference vectors generally characterizes the global topography of the map. In this sense, a globally topographic map is referred to as an ordered state of the SOM. The ordered state is stable because, once attained, this state is not broken during learning.

However, the algorithm frequently forms an undesirable, imperfect topographic map having a topological defect [Fig. 1(b), right]. A topological defect can be defined as a point of global topographic conflict in the map, where the ordered configuration of the reference vectors is separated into multiple, locally ordered regions. The map in such a disordered state relates two similar stimuli to two distant positions on the unit array. This implies that the neighborhood relationships between the stimuli are no longer preserved through this mapping. Such a disordered state of the map is frequently as stable as an ordered state. This indicates that once a topological defect appears in the map, it remains for a long time during learning, resulting in a drastic slowdown of the ordering speed of the map. We note that even under different dimensionality conditions, other types of topological defects still appear and slow down the ordering process [Figs. 1(c) and 1(d)].

In practical situations, ordered topographic maps should form quickly and robustly. In general, the appearance of a topological defect depends on the initial values of the reference vectors and the temporal order in which the stimuli are presented [Fig. 1(e)]. Controlling these two conditions may allow the formation of topographic maps without an interim defect. However, such an approach is neither practical nor easy, and thus, a different solution must be sought. In this context,

some of the authors have reported that when asymmetry is introduced in the neighborhood function $h(r)$, topological defects can be effectively moved out of the map [13]. As a result, the correct map formation can be accelerated. The asymmetric neighborhood function [Fig. 2(b)] is written as

$$h_\beta(\tilde{r}_{ic}) = 2\left(\frac{1}{\beta} + \beta\right)^{-1} \exp\left(-\frac{\tilde{r}_{ic}^2}{2\sigma^2}\right), \quad (3)$$

where the asymmetrically rescaled distance r_{ic} is

$$\tilde{r}_{ic} = \begin{cases} \sqrt{\left(\frac{\mathbf{r}_{ic} \cdot \mathbf{k}}{\beta}\right)^2 + \|\mathbf{r}_\perp\|^2}, & \text{if } \mathbf{r}_{ic} \cdot \mathbf{k} \geq 0, \\ \sqrt{(\beta \mathbf{r}_{ic} \cdot \mathbf{k})^2 + \|\mathbf{r}_\perp\|^2}, & \text{if } \mathbf{r}_{ic} \cdot \mathbf{k} < 0. \end{cases} \quad (4)$$

A unit vector \mathbf{k} indicates the asymmetry direction on the array of units $\mathbf{r}_\perp \equiv \mathbf{r}_{ic} - (\mathbf{r}_{ic} \cdot \mathbf{k})\mathbf{k}$, and the asymmetry parameter $\beta \geq 1$ represents the degree of asymmetry. The symmetric function can be derived from the asymmetric function by substituting $\beta = 1$.

In the next section, we compare the ordering process induced by the symmetric and asymmetric neighborhood functions and show that the asymmetric neighborhood function successfully accelerates the ordering process under broad conditions. For numerical simulations, the stimulus vector \mathbf{x} is randomly sampled from a uniform distribution $[0, 1]^{D_s}$, unless otherwise noted. To quantitatively measure the ordered state of a map, we introduce an order parameter [Fig. 3(a)]. A topographic map corresponds to a globally ordered arrangement of reference vectors, whereas a disordered map containing topological defects is separated into multiple, locally ordered regions. We denote the size of each locally ordered region by S_i and define the order parameter as $\eta \equiv \frac{\max_i \{S_i\}}{\sum_i S_i}$ ($0 < \eta \leq 1$). The order parameter $\eta = 1$ corresponds to a correct topographic map. The value of the order parameter decreases as the map is further divided into locally ordered regions. We describe the calculation of S_i in the Appendix.

We also introduce a distortion parameter to quantify the degree of distortion of a map [Fig. 3(b)]. The previous study by some of the authors [13] showed that when asymmetry is

introduced to a neighborhood function with a fixed intensity and direction, the generated map tends to be distorted. A distorted map does not correctly represent the stimuli distribution and is less useful for practical data processing. To measure the distortion of the map, we define the distortion parameter as $\chi \equiv \frac{\sqrt{\text{Var}[\Delta_i]}}{\text{E}[\Delta_i]}$, where Δ_i denotes the size of the Voronoi cell formed by the reference vector \mathbf{m}_i . Because we consider uniform stimulus distribution, the Voronoi cell size of a unit is proportional to the probability that a given stimulus is mapped to the unit. For more uniform maps (with less distortion), the variation of the Voronoi cell sizes is reduced, where χ takes lower values. Note that χ is meaningful only if the map is globally ordered (i.e., $\eta = 1$).

III. RESULTS

A previous study by some of the authors showed that the asymmetric neighborhood function can quickly eliminate topological defects under the condition that both the stimulus space and the array of units are one-dimensional (i.e., $D_s = D_u = 1$) [13]. However, such a simple scenario, under which analytical investigations are possible [8,14], may be irrelevant with respect to practical application for the following reasons: First, a one-dimensional stimulus or data space ($D_s = 1$) indicates that the dimension of stimulus or data space cannot be reduced any further. Second, the SOM algorithm generally maps the high-dimensional data set to a low-dimensional unit space, which implies that essential information is extracted from the data. Thus, in practice, the dimensionality condition should be $D_s > D_u$, unlike that of a simple, one-dimensional condition.

Because the properties of topological defects vary for different values of D_s or D_u , it is not evident that the asymmetric neighborhood function is always effective. Thus, we investigate whether the asymmetric neighborhood function can eliminate topological defects when the dimensions of D_s and D_u are varied, as summarized in Fig. 4. We start with low values and gradually increase each value of D_s and D_u such that $D_s \geq D_u$. We consider the cases where $D_u = 1$ or 2 because mapping the stimulus space to more

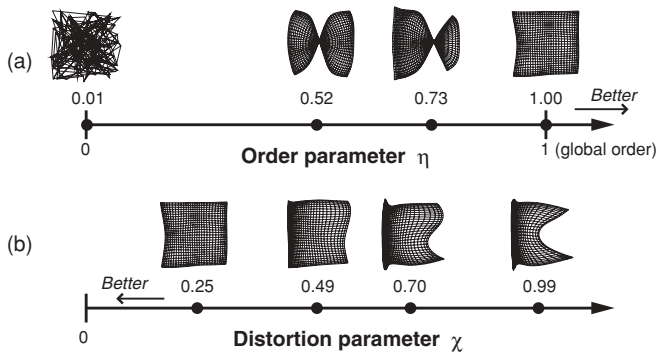


FIG. 3. Typical states of a map for $D_s = D_u = 2$ and corresponding values of the introduced parameters. (a) Order parameter η . $\eta = 1$ for a globally ordered mapping. Value of η decreases as topological defects separate the map into multiple, locally ordered regions. (b) Distortion parameter χ . Large χ corresponds to a strongly distorted map that represents stimulus distribution less accurately than nondistorted maps.

$D_s \backslash D_u$	1	2	3,4,...
1	 Effective (see [1])	 Effective (see section IIIB)	Effective for low-dimensional stimulus distribution (see section IIIC)
2	(Excluded, because $D_s < D_u$)	 Effective (see section IIIA)	

FIG. 4. Condition matrix for various dimensionalities of stimulus space D_s and unit array D_u . This study argues that the asymmetric neighborhood function is effective for removing topological defects for $D_s = 2$ (Secs. III A and III B) and for $D_s \geq 3$ (Sec. III C). The case $D_s = D_u = 1$ was previously studied in Ref. [13]; the condition $(D_s, D_u) = (1, 2)$ is excluded because $D_s < D_u$.

than a two-dimensional unit space would be less useful for data processing. First, we confirm that the asymmetric neighborhood function is effective for $(D_s, D_u) = (2, 2)$ and $(2, 1)$. We then extend the discussion to higher-dimensional cases, where it is observed that the asymmetric function is effective. Thus, the practical use of the algorithm is a realistic possibility. Toward the end of the section, we show the results when the algorithm is applied to real-world data sets, which demonstrates that the map formation process can be improved for practical applications.

A. Dimension-preserving mapping

First, we investigate the case in which the stimuli are two-dimensional and the mapping preserves dimensions ($D_s = D_u = 2$). This condition corresponds to the simplest case of a unit array in a two-dimensional lattice, which is frequently used in practice. We set the initial reference vectors such that the map has a single topological defect at the center, and we focus on its behavior during the learning process. When the conventional symmetric neighborhood function is used, a topological defect fluctuates around the initial position but remains stable for virtually an infinite number of time steps. We confirm that a defect is not removed until $t = 10^7$. This is contrary to the case $D_s = D_u = 1$ in which a defect moves similarly to a random-walk particle and disappears within finite time steps. This suggests that a disordered state in the two-dimensional SOM is more robust than in the one-dimensional SOM.

The asymmetric neighborhood function expressed in Eq. (3) eliminates the topological defect much faster than the symmetric function [Fig. 5(b)], because the asymmetry induces a drifting of the defect. We also confirm that the removal of the defect occurred reliably, over many trials [Fig. 6(a)]. The order parameter η averaged over 30 trials quickly attains the value 1 using the asymmetric function, indicating that in all

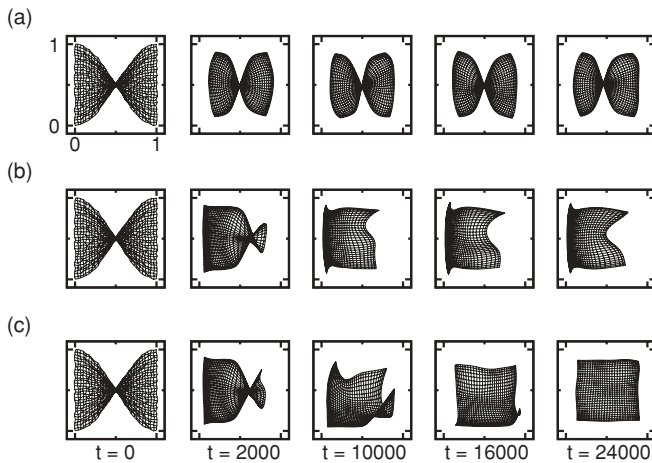


FIG. 5. Snapshots of the typical ordering process of the SOM using (a) symmetric, (b) fixed asymmetric, and (c) modified asymmetric neighborhood functions. Both the stimulus space and the array of units are two-dimensional ($D_s = D_u = 2$). As an initial condition, the map has a single topological defect at the center. The parameters of the simulation are $N = 30 \times 30 = 900$, $\alpha = 0.05$, $\sigma = 4$, $\beta = 1.5$, $t_{\text{sym}} = 24\,000$, and $\omega = \frac{\pi}{24\,000}$.

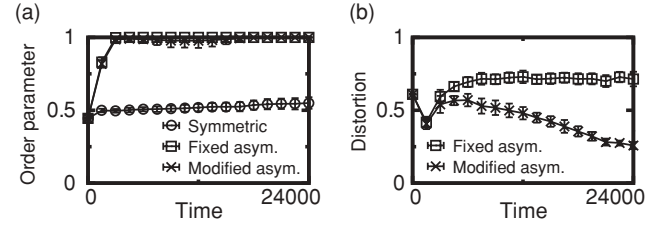


FIG. 6. Time development of (a) order parameter η and (b) distortion parameter χ , averaged over 30 trials under the same condition as that in Fig. 5. The standard deviations are plotted as error bars.

trials the topological defect is removed and the map becomes globally ordered. In contrast, using the symmetric function, the averaged value of η is approximately 0.5, indicating that the defect remains at the center of the map and divides the map roughly in half, into two locally ordered regions.

Although the maps obtained with the asymmetric function have no topological defects, they tend to be distorted and are still not strictly topographic. Because we employ uniform stimulus distribution, the reference vectors corresponding to the correct topographic mapping are uniformly distributed in the stimulus space. However, the resultant maps are distorted [Fig. 5(b)]; i.e., the reference vectors are located in the stimulus space with a bias and do not correctly represent the stimuli distribution. To reduce the map distortion, we further modify the method such that the form of the neighborhood function changes with time as follows [Fig. 7(a)]: First, as the learning proceeds, to average out the distortion of the map, the direction of asymmetry \mathbf{k} is rotated at angular velocity ω as $\mathbf{k}(t) = (\cos \omega t, \sin \omega t)^T$. Second, to reduce distortion, the asymmetry parameter β is gradually decreased as

$$\beta(t) = \begin{cases} 1 + (\beta_0 - 1)(1 - \frac{t}{t_{\text{sym}}}), & \text{if } t < t_{\text{sym}}, \\ 1, & \text{if } t \geq t_{\text{sym}}, \end{cases} \quad (5)$$

where t_{sym} is the time at which the neighborhood function is symmetric ($\beta = 1$). The asymmetric neighborhood function with this modified method is called the modified asymmetric function, while that without the modification is called the fixed asymmetric function.

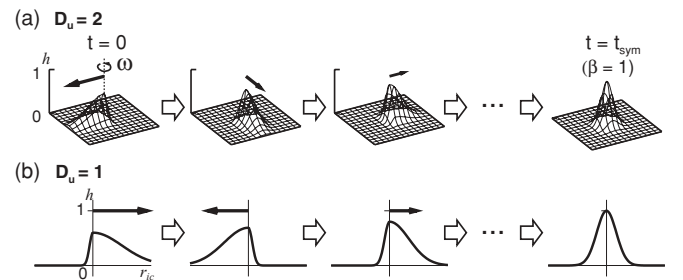


FIG. 7. Modified method to reduce map distortion. (a) When $D_u = 2$, the direction of asymmetry \mathbf{k} (indicated by the direction of the arrows) is rotated at angular velocity ω , and, simultaneously, the asymmetry parameter β (length of the arrows) is gradually decreased as the neighborhood function becomes symmetric ($\beta = 1$) at $t = t_{\text{sym}}$. (b) For $D_u = 1$, the asymmetry direction \mathbf{k} is inverted at every T_{flip} time steps, and the asymmetry parameter β is gradually decreased.

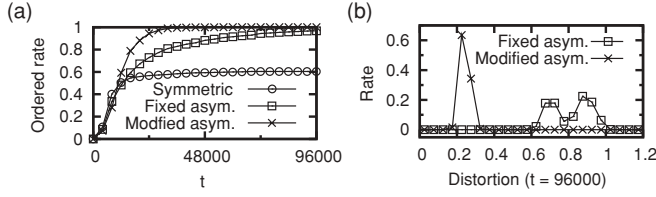


FIG. 8. By using random initial reference vectors, the asymmetric neighborhood function reliably generates an ordered map, and the modified method effectively reduces distortion ($D_s = D_u = 2$). (a) Number of trials in which the map is ordered ($\eta = 1$) (ordered rate). (b) Distribution of the distortion parameter χ at $t = 96\,000$. The modified method effectively reduces the distortion of the map. The parameters are the same as those in Fig. 5, except for $t_{\text{sym}} = 96\,000$.

By using the modified asymmetric function, the emergence of topological defects and distortions of the map can be simultaneously suppressed [Fig. 5(c)]. We also confirm that the distortion reduction occurred in a reliable manner, as confirmed over many trials [Fig. 6(b)]. The distortion parameter χ predictably decreased by using the modified method, as shown by averaging its results.

Until now, we have investigated the cases in which the map has a single topological defect at the center. However, in general, multiple topological defects may emerge within a map. To examine the effectiveness of the asymmetric neighborhood function in such cases, the initial reference vectors are randomly selected from the uniform distribution within $[0, 1]^2$. We perform 1000 trial simulations and examine the rate of trials in which the map is ordered (ordered rate) at each time step and the distribution of distortion parameter χ at the end of the learning process. By using the symmetric function, the ordered rate reaches the plateau value of approximately 60%, indicating that defects remain at the end of the learning process in 40% of the trials [Fig. 8(a)]. The modified asymmetric function can order the map in all trials and effectively reduce the distortion of the map simultaneously [Fig. 8(b)]. In contrast, the growth of the ordered rate is slower using the fixed asymmetric function. This is because the intensity of the drift force depends on the directions of asymmetry of the neighborhood function and the topological defect. This indicates that the fixed asymmetric function has its *strong* and *weak* directions with respect to eliminating a defect. Because the random initial condition allows topological defects to emerge in various directions, the ordering speed obtained using the fixed asymmetric function slows down if a defect emerges in the weak direction.

B. Dimension-reducing mapping

Next, we examine the case in which two-dimensional stimuli are mapped to a one-dimensional array of units (D_s, D_u) = (2, 1). This is the simplest scenario for dimension-reducing mappings. A topological defect is in the shape of a cross-point formed by the chain of reference vectors. We set the initial reference vectors such that the map has a single topological defect. It is difficult to remove the defect by using the conventional symmetric neighborhood function [Fig. 9(a)]. Similar to the case when $D_s = D_u = 2$, for (D_s, D_u) = (2, 1),

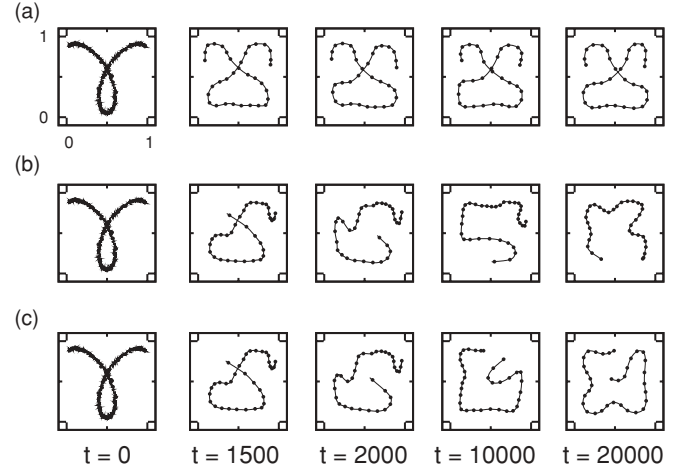


FIG. 9. Snapshots of typical ordering process of the algorithm using (a) symmetric, (b) fixed asymmetric, and (c) modified asymmetric neighborhood functions, where two-dimensional stimuli are mapped to a one-dimensional chain of units ($D_s = 2, D_u = 1$). As an initial condition, the map has a single topological defect at the center. To visualize the distortion of the map well, the reference vectors of every 30 units are depicted as filled circles. The parameters are $N = 1000$, $\alpha = 0.05$, $\sigma = 30$, $\beta = 2$, $t_{\text{sym}} = 20\,000$, and $T_{\text{flip}} = 5000$.

a defect is stable and remains in the map for virtually an infinite number of (at least 10^7) time steps.

We examined the fixed and modified asymmetric neighborhood function and obtained results similar to those for $D_s = D_u = 2$. We summarize the methods and the results as follows:

- (i) A fixed asymmetric function can quickly remove cross-point topological defects, but tends to distort the map [Fig. 9(b)].
- (ii) To reduce the distortion of the map, we introduce the modified method in which the asymmetry parameter β is gradually decreased as in Eq. (5), and the direction of asymmetry is inverted at every T_{flip} time steps [Fig. 7(b)]. A correct topographic map can be generated without distortion by using the modified asymmetric function [Fig. 9(c)].
- (iii) Even for random initial reference vectors, correct topographic maps without defects and distortion can be reliably generated using the modified asymmetric function (Fig. 10). Unlike the case $D_s = D_u = 2$, the growing speed of the ordering rate is similar for fixed and modified asymmetric

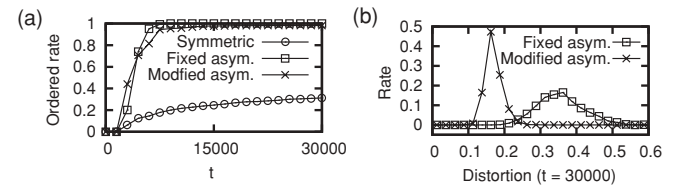


FIG. 10. By using the random initial reference vectors, the modified asymmetric neighborhood function generates an ordered map in most of the 1000 trials ($D_s = 2, D_u = 1$). (a) Time course of the ordered rate. (b) Distribution of the distortion parameter χ at $t = 30\,000$. The parameters are the same as those in Fig. 5 except for $t_{\text{sym}} = 30\,000$.

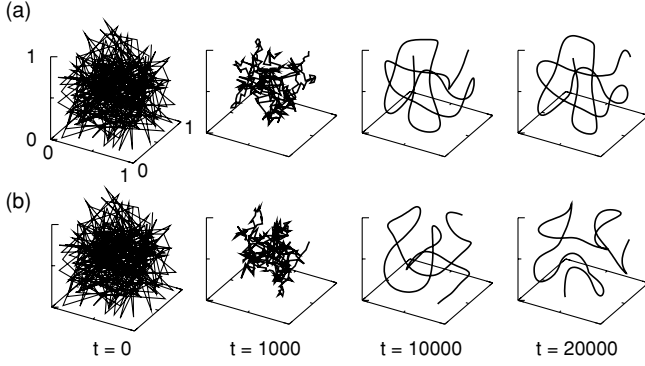


FIG. 11. When uniformly distributed stimuli in three-dimensional space are mapped to a one-dimensional chain of units, $(D_s, D_u) = (3, 1)$, no topological defects emerge even if the conventional symmetric neighborhood function is used. No significant difference can be seen between the learning process using (a) symmetric and (b) modified asymmetric neighborhood functions. Initial reference vectors are randomly selected from $[0, 1]^3$, and the parameters are $N = 300$, $\alpha = 0.05$, $\sigma = 5$, $\beta = 3$, $t_{\text{sym}} = 20\,000$, and $T_{\text{flip}} = 5000$.

functions [Fig. 10(a)] owing to the lower degree of freedom in the defect direction.

C. Higher-dimensional stimuli

Finally, we investigate the case in which the stimulus space is more than two-dimensional. For practical applications of the SOM algorithm, the dimension of stimuli or data is usually much higher than that of the array of units. Thus, this scenario is more relevant to the application of the algorithm than our two cases. In this scenario, we examined two sample cases for $(D_s, D_u) = (3, 1)$: First, the stimuli are uniformly distributed in the cubic region $[0, 1]^3$, and, second, the distribution is restricted to the vicinity of a two-dimensional curved surface.

The uniform distribution of the stimuli ensures that no topological defects are formed during learning, even when the conventional symmetric neighborhood function is used (Fig. 11). This is because when a one-dimensional chain of units fits three-dimensionally distributed stimuli, topological defects (cross-points) cannot emerge because of the high dimensionality of the stimulus distribution. Accordingly, in this case, there is no significant difference between the maps generated using the symmetric and asymmetric neighborhood functions with respect to the emergence of topological defects. A similar situation occurs in the embedding of trajectories of dynamical systems. It was shown that when a d' -dimensional trajectory is embedded in a d -dimensional state space, the condition $d \geq 2d' + 1$ is sufficient to have no generic self-intersections of the trajectory [15]. Analogously, no topological defects emerge and stay within the map during the learning process if $D_s \geq 2D_u + 1$. It may seem that it is no longer necessary to introduce asymmetric neighborhood functions when this condition is satisfied. Indeed, in our sample case $(D_s, D_u) = (3, 1)$, the condition is satisfied. Nevertheless, we believe that our modifications to the SOM algorithm are still relevant when the following aspects are considered. We have assumed that the stimuli are distributed uniformly in

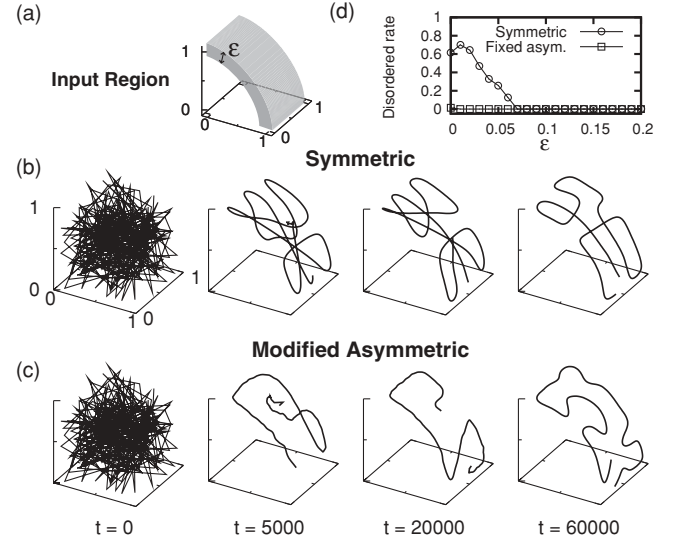


FIG. 12. (a) Stimulus distribution limited to the vicinity of a two-dimensional surface in the three-dimensional space. (b) Emergence of topological defects when the symmetric function is used. (c) Use of the asymmetric function resulting in correct map formation. The parameters are the same as those in Fig. 11, and the thickness of the stimulus distribution is $\epsilon = 0.05$. (d) Rate of trials (70) in which the map has topological defects at $t = 60\,000$. Using the symmetric function, topological defects occur for small ϵ .

a three-dimensional region. However, this assumption is not true for many practical applications. In the application of the algorithm, we aim to extract topographic information about the low-dimensional nonlinear manifold on which the stimuli (or data) are distributed. This means that even if the dimension of the stimuli is high, the data are typically distributed along a lower-dimensional manifold.

Accordingly, to examine the effectiveness of the asymmetric neighborhood function in such scenarios, we consider another sample case in which the stimulus distribution is limited to the vicinity of a two-dimensional curved surface in the three-dimensional stimulus representation space (Fig. 12). As a result, cross-point topological defects occur when the symmetric neighborhood function is used, as in the case of uniformly distributed two-dimensional stimuli (e.g., Fig. 9). When the asymmetric neighborhood function is used, topological defects are removed and a correct topographic map can be obtained. If the thickness of stimulus distribution ϵ is gradually reduced, below a particular value of ϵ , there is a significant increase in the tendency that the topological defects remain after learning [Fig. 12(d)]. We evaluate the rate of trials in which topological defects remain at the end of the learning (disordered rate). We observe that for small ϵ (≤ 0.07), the disordered rate is a nonzero value when the symmetric function is used. In contrast, the disordered rate is zero over the entire range of ϵ when the asymmetric function is used, as shown in Fig. 12(d), implying that topological defects can be removed for any ϵ .

To verify whether the asymmetric function can have practical advantages, we also consider three real-world data sets provided by the UCI Machine Learning Repository [16]. The first is the *Auto MPG* data set, which contains 398 automobile

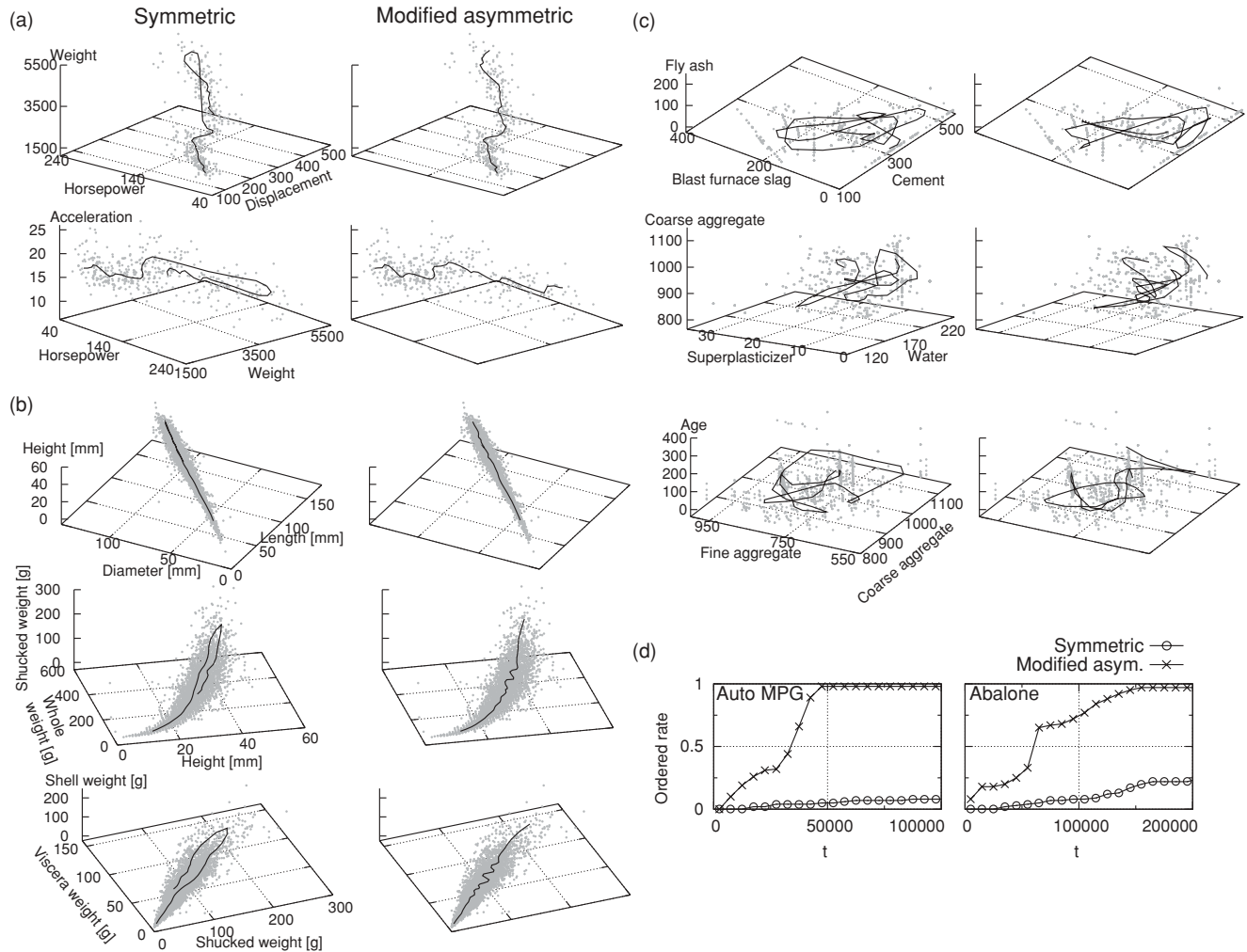


FIG. 13. Results of the application of SOM to three real-world data sets. (a) Snapshots of typical learning results for the Auto MPG data set. While the symmetric function typically results in maps having kinklike defects (left column), the modified asymmetric function successfully generates globally ordered maps (right column). The data are depicted as gray points. Different three-dimensional subspaces of the data space are shown in each row. (b) Same as (a), except for the Abalone data set. (c) Same as (a), except for the Concrete Compressive Strength data set. In this case, because the data points are not distributed along any apparent low-dimensional manifold, the generated maps are randomly located along the input space and are less useful for data processing for both the symmetric and asymmetric neighborhood functions. (d) Time development of the average ordered rate calculated over 100 trials for the Auto MPG and Abalone data sets. For both data sets, the modified asymmetric function quickly attains a high ordered rate, indicating that the globally ordered map is reliably generated. In each trial, different initial reference vectors and data presentation order are used. Simulation parameters are $N = 45$, $\alpha = 0.05$, $\sigma = 1$, and $\beta = 1.5$. For the Auto MPG and Concrete Compressive Strength data sets, $t_{\text{sym}} = 100\,000$ and $T_{\text{flip}} = 50\,000$; for the Abalone data set, $t_{\text{sym}} = 200\,000$ and $T_{\text{flip}} = 100\,000$.

specifications. For input data, we select four continuous-valued attributes (i.e., $D_s = 4$) from the original data set, i.e., displacement, horsepower, weight, and acceleration.¹ The second is the *Abalone* data set, which has relevant data regarding the physical measurements of abalone. We use seven continuous-valued attributes for the learning ($D_s = 7$). The third data set is the *Concrete Compressive Strength* data set, which includes quantity of seven ingredients and the age of concrete samples ($D_s = 8$). The data in both the first and second data sets are roughly distributed along a single curved

line (one-dimensional nonlinear manifold) in the input space [Figs. 13(a) and 13(b)]. In contrast, the data in the third set are broadly distributed in the stimulus space, and there is no apparent low-dimensional manifold to be traced [Fig. 13(c)]. As shown in Figs. 13(a) and 13(b), for both the first and second data sets, the modified asymmetric function successfully generates globally ordered maps without topological defects, whereas the symmetric one typically forms folded maps having kinklike defects. This observation demonstrates that the asymmetric function can extract information regarding the low-dimensional manifold of the data distribution more effectively than the symmetric one. As indicated by the ordered state in Fig. 13(d), the probability that the asymmetric function achieves a globally ordered state is significantly larger than that

¹Although the original data set also contains some discrete-valued attributes, we omit them in the present study.

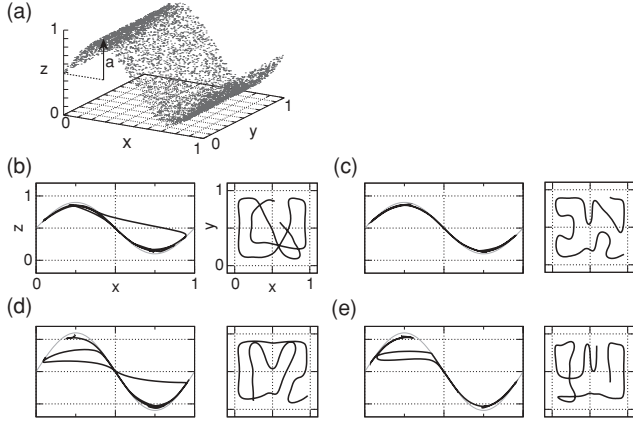


FIG. 14. Sine-shaped stimulus distributions can result in undesirable maps, depending on their curvature. (a) Sine-shaped distribution. The amplitude a controls the curvature of the distribution. (b) Result obtained using the symmetric neighborhood function for $a = 0.4$. The generated maps typically have cross-point defects. The left and right graphs show the projections of the stimulus space to the (x, z) and (x, y) planes, respectively. (c) Typical result obtained using the modified asymmetric function for $a = 0.4$, in which no topological defects are formed, in contrast to (b). At a higher curvature ($a = 0.6$), both the symmetric (d) and modified asymmetric (e) functions typically generate disordered maps and fail to trace the nonlinear manifold, because some shortcut paths away from the true manifold are formed in the map. The parameters are $N = 300$, $\alpha = 0.05$, $\sigma = 5$, $\beta = 3$, $t_{\text{sym}} = 30\,000$, and $T_{\text{flip}} = 5000$.

of the symmetric one. On the other hand, the third data set lacks any apparent low-dimensional structure in its distribution. In this case, as expected, both the asymmetric and symmetric functions similarly generate maps that are randomly located along the high-dimensional input space and are less useful for data processing, as shown in Fig. 11.

For real-world data, our results show that introduction of asymmetric neighborhood functions can facilitate extraction of some low-dimensional structures inherent in the data. However, under the condition that the data are distributed on a highly curved low-dimensional manifold, the SOM using the modified asymmetric function fails to extract the corresponding inherent data structure (Fig. 14). Consider the sine-shaped distributions as shown in Fig. 14(a). The amplitude of the sine shape a can then control the curvature of the distribution. When the curvature is low ($a = 0.4$), the modified asymmetric function generates ordered maps with a high probability [8 of 10 trials; Fig. 14(b)]. In contrast, the symmetric function always fails for the same distribution [Fig. 14(c)]. These results are almost the same as that in Fig. 12. At a higher curvature ($a = 0.6$), all trials with either functions result in undesirable maps that do not correctly trace the manifold [Figs. 14(d) and 14(e)]. These maps have some shortcut paths outside the true manifold, for example, the path from $(x, z) = (0.05, 0.65)$ to $(0.5, 0.5)$ in Fig. 14(d). Therefore, we conclude that even if the asymmetric neighborhood function is used, a highly curved structure inherent in the data tends to hinder the extraction of information from the data distributed on a low-dimensional manifold.

To summarize this section, in the case of high-dimensional stimuli, it is naturally expected that no topological defects

can appear if the condition $D_s \geq 2D_u + 1$ is satisfied and the stimuli are uniformly distributed in a certain high-dimensional region. However, we frequently encounter the situation in which there are some hidden structures in the stimuli or the data, which indicates that the stimuli are distributed along a low-dimensional nonlinear manifold in a large high-dimensional space. In such cases, topological defects inevitably appear in the map but can be removed by using the asymmetric neighborhood function, as in the case $D_s = 1$ or 2. Consequently, the asymmetric neighborhood function can be effectively used to avoid topological defects even in the case of high-dimensional stimuli. On the other hand, when data are distributed along a highly curved manifold, even the SOM algorithm using the asymmetric function can generate an incorrect disordered map in which the map has some shortcut paths outside the manifold.

IV. CONCLUSION

We have examined whether the asymmetric neighborhood function can remove topological defects during the learning process of the SOM algorithm. Although a previous study by some of the authors provided a detailed investigation for the simplest case in which the one-dimensional stimuli are mapped to a chain of units $[(D_s, D_u) = (1, 1)]$, the effectiveness of the asymmetric function under other conditions was not clear. To systematically investigate the effectiveness of the asymmetric function, we gradually changed the dimensionality conditions for the stimulus space and the array of units from low to high values. First, we confirmed that the asymmetric neighborhood function can reliably remove topological defects in the map when both the stimulus space and the array of units are two-dimensional $[(D_s, D_u) = (2, 2)]$ and when two-dimensional stimuli are mapped to a one-dimensional chain of units $[(D_s, D_u) = (2, 1)]$. Moreover, the distortion of maps caused by the asymmetry can be simultaneously reduced by introducing a modified method in which the direction and the degree of asymmetry are changed over time. Second, we examined the case in which the stimulus distribution is limited to the vicinity of a two-dimensional curved surface in a three-dimensional stimulus space. Even in such a case, the asymmetric neighborhood function accelerates the formation of a correct topographic map by efficiently removing topological defects. For practical applications of the SOM algorithm, data are typically distributed along a particular low-dimensional nonlinear manifold. These results suggest that the asymmetric function is also effective for processing high-dimensional data sets. Third, to confirm the effectiveness of the asymmetric function in practical situations, we examined three real-world data sets. The results obtained from the real-world data sets suggest that the introduction of the asymmetric neighborhood function in the SOM algorithm is more effective in achieving a correctly ordered map if there is certain underlying low-dimensional structure in the data distribution. Finally, we found that if data are distributed along a highly curved manifold, even the asymmetric function tends to generate a disordered map that is unsuccessful in extracting the essential low-dimensional structure from the data. In information processing, it is often crucial to extract information regarding a nonlinear low-dimensional

manifold hidden in the data. It is known that sophisticated algorithms, such as locally linear embedding, can successfully solve this problem [17,18]. Our last result indicates that for the SOM algorithm, introduction of only the asymmetric neighborhood function is insufficient to robustly extract the nonlinear structure inherent in the data. A variant of the SOM was proposed to overcome the high nonlinearity of the manifold [19]. A combination of such an extended algorithm and the asymmetric neighborhood function is of great interest. However, a more complete study of this topic is beyond the scope of this paper and should be studied elsewhere in the near future.

In conclusion, the asymmetric neighborhood function is significant in steadily generating correct topographic maps under a wide range of conditions. This improvement can lead to more reliable applications of the SOM algorithm to complicated, high-dimensional data processing.

ACKNOWLEDGMENTS

This work was supported by KAKENHI (21120002, 19GS2008, 08J05147, 09J04536) from MEXT, Japan.

APPENDIX

To calculate the order parameter $\eta \equiv \frac{\max_l \{S_l\}}{\sum_l S_l}$, we define the size of the locally ordered region S_l as follows: When $(D_s, D_u) = (1, 1)$, the unit of region is an interval between two neighborhood units i and $i + 1$. The possible state of a unit region is increasing ($m_i < m_{i+1}$) or decreasing ($m_i > m_{i+1}$). A locally ordered region is a series of regions in the same state. When $(D_s, D_u) = (2, 1)$, the unit of region is an interval between two neighborhood units i and $i + 1$. On a line connecting the two reference vectors \mathbf{m}_i and \mathbf{m}_{i+1} , if the closest reference vector of any point on the line is unit i or $i + 1$, the region is *nondisturbed*; otherwise, the region is *disturbed*. A locally ordered region is a series of nondisturbed regions. When $(D_s, D_u) = (2, 2)$, the unit of region is a grid formed by the four neighboring units $(i, j), (i, j + 1), (i + 1, j)$, and $(i + 1, j + 1)$. We calculate the vector products $a \equiv (\mathbf{a} \times \mathbf{d})_z$ and $b \equiv (\mathbf{b} \times \mathbf{d})_z$, where $\mathbf{a} \equiv \mathbf{m}_{i+1,j} - \mathbf{m}_{i,j}$, $\mathbf{b} \equiv \mathbf{m}_{i,j+1} - \mathbf{m}_{i,j}$, and $\mathbf{d} \equiv \mathbf{m}_{i+1,j+1} - \mathbf{m}_{i,j}$. A unit region has three possible states: (i) $a > 0 > b$, (ii) $b > 0 > a$, or (iii; twisted) $a, b > 0$ or $a, b < 0$. A locally ordered region is a set of connected regions that are in the same state but are not twisted.

-
- [1] D. H. Hubel and T. N. Wiesel, *The Journal of Physiology* (London) **160**, 106 (1962).
 - [2] D. H. Hubel and T. N. Wiesel, *J. Comp. Neurol.* **158**, 267 (1974).
 - [3] T. Kohonen, *Biol. Cybern.* **43**, 59 (1982).
 - [4] C. von der Malsburg, *Kybernetik* **14**, 85 (1973).
 - [5] A. Takeuchi and S. Amari, *Biol. Cybern.* **35**, 63 (1979).
 - [6] K. Obermayer, G. G. Blasdel, and K. Schulten, *Phys. Rev. A* **45**, 7568 (1992).
 - [7] T. Kohonen, *Neural Networks* **6**, 895 (1993).
 - [8] H. Ritter and K. Schulten, *Biol. Cybern.* **54**, 99 (1986).
 - [9] C. Bishop, M. Svensén, and C. Williams, *Neural Comput.* **10**, 215 (1998).
 - [10] T. Kohonen, *Self-Organizing Maps*, 3rd ed. (Springer, Berlin, Heidelberg, New York, 2001).
 - [11] T. Geszti, *Physical Models of Neural Networks* (World Scientific, Singapore, 1990).
 - [12] R. Der, M. Herrmann, and T. Villmann, *Biol. Cybern.* **77**, 419 (1997).
 - [13] T. Aoki and T. Aoyagi, *Neural Comput.* **19**, 2515 (2007).
 - [14] E. Erwin, K. Obermayer, and K. Schulten, *Biol. Cybern.* **67**, 35 (1992).
 - [15] F. Takens, in *Springer Lecture Notes in Mathematics*, Vol. 898 (Springer, New York, 1980), p. 366.
 - [16] A. Asuncion and D. J. Newman, UCI Machine Learning Repository (2007) [<http://archive.ics.uci.edu/ml/>].
 - [17] J. B. Tenenbaum, *Advances in Neural Information Processing Systems* (NIPS) **10**, 682 (1998).
 - [18] S. T. Rowies and L. K. Saul, *Science* **290**, 2323 (2000).
 - [19] H. Yin, *Neural Networks* **21**, 160 (2008).